Enhancing Animal repellent system using AI and Deep Learning

Kasthuri U[1*], Priyadharshini P[2], Dhanush R[3], Kavin Kumar P S[4], Madhu Nandha G[5] and Riyaz MD[6]

[1, 2]Assistant Professor, Department of Computer Science and Engineering, KSR College of Engineering, Tiruchengode-637215, Tamil Nadu, India.
[3, 4, 5] Research Scholar, Department of Computer Science and Engineering, KSR College of Engineering, Tiruchengode-637215, Tamil Nadu, India.
kasthuri@ksrce.ac.in

Abstract. The goal is to design a real-time animal incursion detection system based on the YOLOv5-based model to promote wildlife monitoring using image data and accurately detect animal presence. This technique utilizes deep learning along with YOLOv5 architecture known mostly for its efficient performance in object-identifier tasks. As the structured format of the data reflects on training stage, the very first thing to do here is creating the dataset (that zip file), which includes unzipping and ordering the picture files. During the model training process, the model is fine-tuned with the following important parameters to achieve the best detection performance on epoch, batch size and image size. A configuration file called model. The yaml file basically tells the system how to train so it can detect thousands of different types of animals. Then, the machine learning system is trained over a handful of well-chosen samples of the images and, depending on a confidence threshold, a carefully filtered detection for an animal incursion is identified. Here, we visualize the results with OpenCV and Matplotlib which offers a comprehensive method of evaluating the accuracy of the model. Such a multi-faceted approach combines cutting-edge computational techniques to facilitate scalable and efficient detection and characterization of animal behavior across different settings.

Keywords: Wildlife monitoring, animal detection, yolov5 model, deep learning techniques.

1 Introduction

Wildlife monitoring is vital to successful ecosystem management, understanding animal behavior and addressing challenges including habitat preservation and reduction of human-wildlife conflict. In this project, an animal incursion detection system will be developed using a state-of-the-art deep learning YOLOv5 model which is an efficient object detection model with a weighted average score. The idea is to train a model to capture animal occurrences in photos with a balanced data set across different groups of animals. During the training phase, the settings are tuned to achieve the best result: parameters such as image size, batch size or number of epochs. A well thought out data set ensures that the pictures are structured to facilitate training the model. The usefulness of models in detecting various species of animals is evaluated and visualised with the use of data science tools, such as OpenCV / Matplotlib. The results of this project demonstrate the potential for applying ML techniques to improve monitoring protocols, and improve conservation and ecosystem management strategies at the same time.

2 Literature review

A work by Alexander Buslaev et al. proposed a new data augmentation, which is an effective method used to complement matching input-output pairs, augment the number

© The Author(s) 2025
S. Kannadhasan et al. (eds.), Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 24), Advances in Computer Science Research 120,
https://doi.org/10.2991/978-94-6463-718-2_93

of samples, thus increasing the diversity of labeled training sets. Image augmentations are widely used for performance improvements in computer vision and have become common as an implicit regularization method to prevent overfitting on deep learning models. While most deep learning frameworks expose basic image manipulations, the list is usually short to certain rendering (flipping, rotate, scale, and crop). Different image augmentation libraries can also have vast differences in the speed at which they process the image. Albumentations is an open source fast, flexible image augmentation library for machine learning. We do this in order to explain the main features and motivation of Albumentations, as well as its design principles that greatly influenced its design. We also show that in comparison to most popular image augmentation libraries, Albumentations performs better in most of the image transform operations and we state use-cases of image augmentations for various computer vision applications. Deep artificial neural networks and other modern machines learning models often need to have millions of parameters and learn to generalize decently well on massive annotated databases [1].

According to Research of Jaqueline J. Brito et al., The computational methods revolutionized the modern biology. Computational technologies are becoming evermore entrenched in the biomedical community, yet open data, open software and reproducibility is applied in an inconsistent and distributed manner across academic institutions, funders and publishers. Publications may describe academic software for which the relevant resources (source code, documentation) do not exist or become unavailable. Such information further jeopardizes peer review's ability to critically assess a publication's technical quality and scientific contribution. Often, this work created by secondary work is restricted to academic software where some extra information is missing, resulting in a narrow scope of work. In the interest of encouraging reproducibility, transparency, and rigor in computational biology — exactly the principles to be emphasized within life science programs — we provide four domains of eight proposals. We hope that our recommendations for improvement in software usability, availability, and archival stability will serve as the basis for a sustainable data science ecosystem in life science and biomedical research. Biomedical informatics contributes more and more to the development of processes that foster open data, open software and reproducible research in the academic world. When researchers share all research materials publicly — in a way that is discoverable and archivally stable, from raw data through installable packages and source code — it enables the computation of already-published findings. But those publications are threatening reproducibility, transparency and scientific rigor [2], by withholding data and source code.

Janusz Kacprzyk et al. in this article Advances in Intelligent Systems and Computing, The "Advances in Intelligent Systems and Computing" series focuses on the theory, applications, and design methodologies of intelligent systems and intelligent computing. Practical Skills includes advice on just about every field: engineering, natural sciences, computer and Information technology (ICT), economics, business and e-commerce, the environment, healthcare, and life sciences. Topics of interest include (but are not limited to) any domain of computational intelligence; soft computing and its applications (neural networks, fuzzy systems, etc.); cognitive science and cognitive systems; artificial life; virtual world and society; cognitive neuroscience; perception and vision; DNA and immune-based systems; self-organizing and adaptive systems; e-learning and teaching; human-centered and human-centric computing; recommender systems; intelligent control; robotics and mechatronics (including human-machine teaming); knowledge-based paradigms; learning paradigms; machine

ethics; intelligent data analysis; knowledge management; intelligent agents; intelligent decision making and support; intelligent network security; trust management; interactive entertainment; Web intelligence; and multimedia.

Pritul M. Dave et al. in the current work. The increase in cars and traffic congestion is a global problem that has been suggested However, one of the problems again with the traffic management is the traffic signal timing is not dynamic. “As such, it often requires a driver to linger longer on the road even in the absence of any other vehicles, which leads to unnecessary wait times due to idling cars, as well as needless fuel consumption and emissions.” In their research design, intelligent traffic management systems re-frame relatively few existing technologies such as digital image processing, time series forecasting, IoT, etc. Computer vision is a new idea for the intelligent control of traffic flow. Thus, it is proposed a progressive approach for instantly optimize the signal of traffic, that predicts the sufficient time to also clear the line using ML and DL techniques. The paper investigates the two-stage approach. The other Begins tallying count of vehicle class Once You Looked at the time frame is the active object detection phase. Concentrating on second phase, recommended one method relying on ensemble learning technique using eXtreme Gradient Boosting to identify the optimal green light window time when traffic light turns into green light. Additionally, the proposed technique is compared against other implementations of YOLO and prediction algorithms. Experimental research indicates that YOLOv4 with XGboost algorithm returns the best result accurately with minimum trade-off between accuracy and inference time [4].

Deepak Patil et al. There is a powerful and often lethal competition between India’s animals and its expanding population. This results in a range of harmful outcomes including injury, death, destruction of crops, destruction of human property, threat to livestock, destruction of habitat, and much more. For decades, it has been shown that measures to protect habitat, such as erecting electric fences, building trenches, monitoring herbivores directly, using dogs as guards, etc., are temporary, impracticable, and pose hazards for people and animals alike. We need a mitigation strategy — dealing with this problem now and moving toward protecting both human and wild animals in the future. There are several animal watch and deterrent systems out there with IoT connectivity, however maybe a touch of artificial intelligence will help them be more effective and get to the limits where either the IoT or traditional security can go. A proposed system that helps take care of people and live-stocks on the edge of forest area and minimize the disastrous effects of wildlife-human conflicts by building in an automatic mechanism that will track their presence and pushed to the forest area, a wildlife-humans threat, without causing loss of life. About 500 species of animals can be discerned in the 708,273 km2 of forest that comprise about 21.54% of India’s land area. Urbanization is one of the major contributing causes of forest-land clearing in the long term [5].

3 Related work

Wildlife monitoring and analysis research has taken place for the past few decades. We focus on finding animals in natural landscapes through camera-trap networks for wildlife monitoring and analysis in this research. Camera-trap sequence data are very crowded, making animals hard to spot; hence detection is poor, leading to high rates of false discovery. To resolve this problem, we used a camera-trap database comprising candidate animal suggestions using multilayer graph cuts in the spatiotemporal paradigm. The clues are used to create a verification process that checks if a given

patch is background or animal. We developed an Animal recognition model based on the self-learned DCNN features. We then use modern machine learning methods (ensemble trees, support vector machines, k-nearest neighbors, etc.) to classify this efficient feature set. Our extensive results show that our detection model using DCNN features achieves 91.4% accuracy on a common camera-trap dataset.

4 Methodology

The proposed system architecture is implemented on YOLOv5 used for object detection, and it enabled animals to detect successfully in different environments. This is done using a carefully selected dataset with images of different species of animals, improving the detection accuracy and model generalization. The image size, batch size, epoch count, and other hyper parameters in the training phase are tuned to maximize the system for specific detection tasks. After trained model recognizes animals when given input photos, detection process generates coincidence levels outputs. For result presentation we embedded user-oriented visualization module of the system for displaying detection results on the input images: user-oriented interface facilitates inspection of results and validation of the accuracy of the system. Fig.1 shows the System flow diagram.

4.1 Load data

This module is all about ingestion, i.e., Importing the image dataset from storage to organize and retrieve the data in the pipeline. We are grouping the images into predefined classes for better efficiency in terms of processing and for maintaining a standard structure of the dataset. We also validate data integrity in this stage where we look for corrupt or missing files. The module also ensures that along with this corresponding label are generated correctly in a specific annotation format required in order to train the model, since the YOLOv5 models heavily depends on correctly labelled annotations during the training phase in order to make efficient predictions.

4.2 Pre-processing

This module changes the dataset in multiples ways so that input for the model can be standardized. To ensure that it looks the same throughout the entire collection, images are resized to the input dimensions of the YOLOv5 architecture. Internally, the photo is converted to have pixel values that minimize the impact of changing lighting conditions and reduce processing. To prevent the model from overfitting to the dataset, they performed Data augmentation techniques such as flipping, rotation, and scaling are used to expand the number of samples available for training the model. These changes are made in the annotations associated with the images, keeping the training process correct.

4.3 Feature extraction

In this module, we will be using the YOLOv5 network to identify and extract items from the images. Convolutional layers are simply many filters that can look at images at different scales to learn local and global high-level features. In order to improve image detection of objects with a variety of sizes and forms, this module also adds multiscale feature gathering algorithms. It creates multidimensional feature maps that allow the model to recognize multiple

4.4 Training and testing model

In the following section of the module, it will train the YOLOv5 model using a pre-processed dataset in a supervised learning method. The training phase is primarily aimed at reducing the loss function, which quickly takes shape through one of these iterative algorithms of weight initialization and adjustment-based processes, gradient descent and its core, backpropagation. Hyper parameters (batch size, learning rate, epoch count, etc.) are adjusted to maximize performance. The next phase is testing, in which the studied model is tested using an undetected dataset to see how well the model recognizes as well as classifies objects that the model did not see during training.

4.5 Model evaluation

In this module the YOLOv5 model trained in the previous module is evaluated using a set of metrics. Recall is a measure of a model's ability to find all of the relevant items, while precision is how correct the model's prediction is. Now to balance both these metrics, F1 score is calculated. Some other metrics evaluate how well the predicted bounding boxes overlap the ground truth bounding boxes like Intersection over Union (IoU). These measures of success are especially important since they can tell us where the model performed well and where it could be doing better, providing us with insight for how to improve the training processes in future iterations.

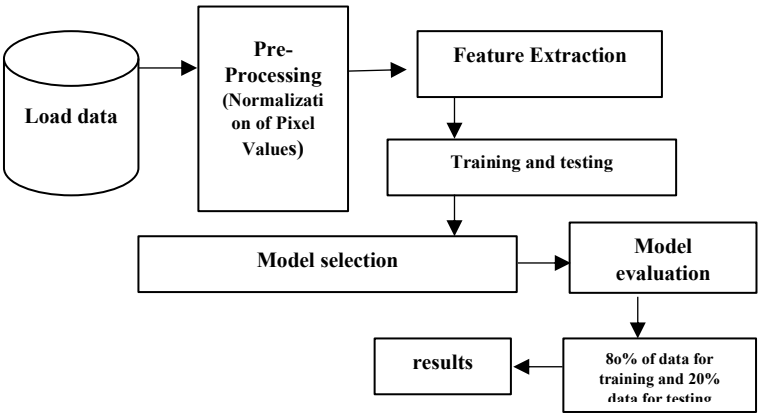

Fig.1. System flow diagram.

5 Algorithm details

The system utilizes the YOLOv5 deep learning architecture which is recognized for its real-time object detection features. The first step is the data preparation; unzipping the input picture collection, naming the folders, and annotating it to comply with the YOLOv5 input format. This ensures that the data is ready for training. This paves way for optimizing the training process of YOLOv5 by way of defining some essential parameters such as image size, batch size, epoch count, etc. These hyperparameters are contained in a YAML configuration file that defines the target animal categories and the architecture of the model. By using convolutional layers, the model learns features from the dataset repeatedly to get to know objects over the course of the training. A confidence level is established in order to reduce false positives and ensure accurate detection. A test dataset, restricted in size, is then used to assess the system after training. Predictions are filtered using the confidence threshold, and the findings are

displayed by overlaying bounding boxes on test pictures with the help of OpenCV and Matplotlib. The algorithm applies non-maximum suppression (NMS) to eliminate redundant detections and ensure only the most accurate predictions are retained. This systematic process results in a scalable and effective approach for detecting invasions of animals in a variety of contexts.

6 Result analysis

The results of the model of YOLOv5 used in this project shows potential of accurately detecting incursions of animals with this configuration on the different test images. Finally, the model's performance was analyzed in terms of accuracy, recall, and f1 score, which are significant measurements to understand how well was the computer able to detect the animals, how many times did it miss detecting them, and how many times did it predict them as a false positive. While recall indicated how well the model did in retrieving all relevant items, precision was more favorable, and this is good, it means that the model correctly identified most of the species in the test data. This performance was also reflected in the F1 score, which provides a balanced picture of accuracy and recall shown in Table 1 and Fig 2. We calculate how well the predicted bounding boxes match with where the object is actually placed using a metric called intersection over union (IoU), which yields a high IoU if our bounding box contains the object well. But there were still some problems, especially when it came to identifying smaller creatures, or more than one creature in an image (like when a hare is only partially visible), indicating areas of the algorithm that could use more work. 2) The system performed well in different picture conditions, signifying generalizability. So further improvements could be made regarding one or two aspects of the model to improve both its accuracy and resilience, but overall, this study has shown that the aspirations the project had when getting started have largely been accomplished.

Algorithm	Accuracy
Existing system	70
Proposed system	85

Table 1. Comparison table.

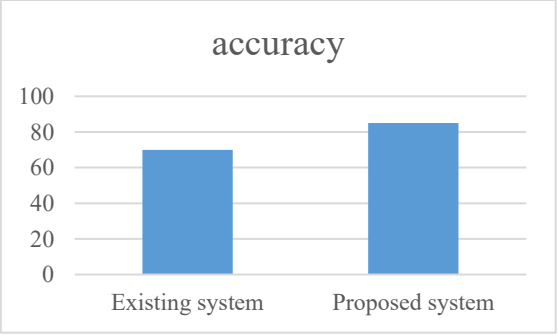

Fig.2. Comparison graph

7 Conclusion

In all, the presented system shows applications of modern deep learning algorithms to accomplish rapid object detecting specifically responsibly on animal intruder alerts. It has been trained and tested regularly, and its performance evaluated, demonstrating that it can effectively detect and classify multiple animal species in images. The implementation of important modules, such as Feature Extraction, Preprocessing, Data Preparation, and Model Validation, has enhanced the whole system performance. More and more hands-on keys the model but there is basic (even with run, the model to high accuracy) In addition to advancing the conversation around wildlife monitoring, this system demonstrates the wide applicability of deep learning to improve detection processes in a multitude of areas. More optimizations can be done for inference to scale it out for more extensive use cases as the demand grows, making it possible to meet even higher requirements.

8 Future work

Eventually, this kind of system will be able to detect a greater diversity of animal species by supplementing the photographs of the animals available by taking more diverse and extensive photographs. Another key focus will be improving the robustness of the model to environmental conditions, such as variable lighting, poor weather conditions, or background clutter. Subsequent enhancement may also focus on higher performance to aid deployment in low resource environment by reducing computation resource consumption. A further improvement is combining us with more advanced detection methods such as multi-object tracking to document movements of animals over time. In addition, the model being studied for other fields of study such as plant disease identifications and monitoring systems of possible applications may result in future research work. As technology advances, new techniques will be developed to process data; therefore, the system design and algorithms will need to change frequently.

The hope is that giving the system a broader and more diverse diet of pictures will help it learn to recognize a wider range of species of animal. Another key focus will be to enhance the robustness of the model to changing environmental conditions, including differing lighting, adverse weather or background clutter. Later progress may also encompass bettering model performance to minimize computing resource needs, which will aid in best effective deployment situations without abundant resources. Another approach to improvement would involve employing advanced detection methods, utilizing an efficient multi-object tracking strategy, to follow the movements of the animals over time. Future works can investigate adapting the model to work on other repair domains: e.g., plant disease detection, monitoring systems, etc. However, as technology and data processing methods advance, it will need to be adapted with more advanced or more adapted algorithms.

References

1. Albumentations: This computer vision technology improves deep convolutional neural networks' performance. https://albumentations.ai/ is the URL (retrieved August 15, 2023)

2. Moore, J.H.; Greene, C.S.; Nogoy, N.A.; Garmire, L.X.; Mangul, S.; Brito, J.J.; Li, J.; Moore, J.H. enhancing software availability, usability, and archival stability to increase repeatability and rigor. arXiv:2001.05127, arXiv 2020.
3. Iglovikov, V.; Rakhlin, A.; Shvets, A.; Kalinin, A.A. Segmenting medical images utilizing pre-trained encoders and deep neural networks. Wani, M.A., Kantardzic, M., and Sayed Mouchaweh, M., eds., In Deep Learning Applications; Springer: Singapore, 2020.
4. Dave, P., Goel, P., Chandarana, A., & Ganatra, A. XGBoost and YOLOv4 together to create a next-generation intelligent traffic control system. e586 in PeerJ Computer Science, 7. (2021).
5. Ansari, N. F., and H. D. Patil. An artificial intelligence-based system for detecting and repelling wild animals. (Pages 291-296) in 2022 International Conference on Computing, Communication, and Power Technology (IC3P). IEEE (2022, January).
6. Saxena A., Gupta D.K., Singh S. "A Deep Learning-Based System for Detecting Animals and Avoiding Collisions." Advances in Communication and Computational Technology, Hura G., Singh A., and Siong Hoe L. (eds.), Springer, 2020.
7. "Deer-Vehicle Collisions Prevention using Deep Learning Techniques," in the Proceedings of the IEEE Cloud Summit in, by T. Fan, R. Sadeghian, and S. Aram. 2020.
8. "Animal Detection using Deep Learning Algorithm," in the Journal of Critical Review, by N. Banupriya, S. Saranya, Rashmi Swaminathan, Sanchithaa Harikumar, and Sukitha Palanisamy. 2020.
9. "Animal Detection in Man-made Environments" by Singh, M. Pietrasik, G. Natha, N. Ghouaiel, K. Brizel, and N. Ray, published in ahead of the IEEE Winter Conference on Applications of Computer Vision (WACV). 2020.
10. Yakub, F., Mamat, N., and Othman, M. F. YOLOv5 Approach for Animal Intrusion Detection in Farming Areas. International Conference on Control, Automation, and Systems (ICCAS), 22nd edition, 2022 (pp. 1–5). IEEE. (2022, November).

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.